Title
Pulling offshore and staying onshore: A framework for analysis of offshoring dynamics

ISBN
9783540888505

Authors
Carmel, E
Dedrick, J
Kraemer, KL

Publication Date
2009-12-01

DOI
10.1007/978-3-540-88851-2_17

Chapter 7
Dynamics in Offshoring

Pulling Offshore and Staying Onshore: A Framework for Analysis of Offshoring Dynamics

Erran Carmel, Jason Dedrick, and Kenneth L. Kraemer

1 Introduction and Motivation

Given that the force of offshoring is one of the most important economic changes in the early 2000s it is vital to understand what propels it further. The landscape of offshoring is such that firms in the wealthy nations (onshore) have already offshored, sometimes extensively. As researchers, we need to ask not whether the firm will offshore, but rather how far the firm will go offshore and what are the subtle factors that are driving this offshore decision process. Therefore, in this paper we propose a framework in order to understand the firm-level decisions that are not as well understood and not as well researched.

There are a number of key variables that we already collectively know about offshoring. First, we already know that offshoring is driven by low wages and large labor pools in India, China and elsewhere. Thus, everything else being equal, work will flow to the less expensive offshore locations.

Second, we already know that there are many enabling factors that "level the playing field," enabling the mobility of projects and tasks. These are the factors that make the playing field almost frictionless. Thomas Friedman (2005) titled his well-known book about offshoring, *The World is Flat*. The flatness of the world is enabled by factors such as technology, standards, collaboration tools, task modularity, and interpersonal relationships (Appendix 1).

Third, we also know that offshore migrations tend to begin with lower-level work and proceed over time to higher-level work. This has been the case for many offshore migrations in many manufacturing fields – from steel to consumer electronics. In the software and IT area, there is considerable evidence that some high-level R&D work now is being moved offshore.

Assuming that the world is flat, we ask: what factors tip an activity from onshore to offshore? What is pulling some activities offshore and what is holding others onshore? To shed light on offshoring and possible tipping points, we were inspired by the landscape of computer hardware to examine Pull Factors. Specifically, we were motivated by Dedrick and Kraemer (2006) who show the Pull Factors in

R. Hirschheim et al. (eds), *Information Systems Outsourcing,*

Taiwan and China for the notebook PC industry. Taiwan "pulled" new product development activities from the U.S., and now some of those activities are being "pulled" to China. In both cases, engineering activities have followed manufacturing to new locations.

It is useful to examine such Pull factors in the context of software and IT services, and to further examine the factors that tend to keep activities onshore, which we call Stay factors. Thus, we define two kinds of factors that shape offshoring decisions:

- *Offshore PULL factors*: Factors in the offshore location that induce more of the work to be tasked offshore.
- *Onshore STAY factors*: Factors in the onshore location that keep the work from migrating offshore. Some have suggested to call these "sticky" factors.

Views about what can be done offshore are constantly changing. While a decade ago one could argue that many high-level, upstream activities such as R&D, product design and software engineering could not be offshored, this position is increasingly difficult to defend today. There is consensus, however, that there are certain activities that are a better fit at offshore locations while others are better to leave in-house and in-country – "onshore." For example, services that require direct physical contact in their delivery must be performed onshore (assuming the final customer is onshore), while theoretically everything else could migrate offshore.

In this paper, we are painting a picture on a large canvas and include all types of software activities. We do not segregate IT from software. This paper is about *any type of software*-related activity: IT services and IT applications, software products, and embedded software.

2 Methodology

Since our motivation was to examine the more nuanced decisions in offshoring, we move in this paper through three stages: framework creation; exploratory data collection and fit to framework; and synthesis.

Our Pull-Stay Framework is based on primary and secondary data through an iterative process of item surfacing, refinement, and regrouping. The initial primary data we used was culled from our many years of cases and interviews across many countries. We also turned to two subdisciplines: the emerging literature on offshoring/outsourcing; and the more traditional literature on "technology transfer."

We used various recent data to examine the robustness of this framework. The data are from multiple sources, multiple firms, and multiple nations: interviews with 12 firms in India during 2005, seven firms in China during 2006; and five firms in the U.S. from 2006–2007. Since all our data collection are anonymous, we use aliases when discussing the firm names. U.S. firms with aliases include Printco, PDAco, MP3co, StorageCo, US-Tech; Chinese firms with aliases include C-Soft. Some secondary data sources are also used.

3 The Pull/Stay Factors and Framework

We identified nine pull/stay factors from existing literature and research (Table 1). It is important to note that many of the factors can be either Pull or Stay, or simultaneously both. Crown Jewels is the only factor that is unambiguously a STAY factor. There are two unambiguous PULL factors: Freshness and Eagerness.

Each of the factors is defined and discussed in this section.

Tacit Knowledge. Tacit knowledge is that which is difficult to write, document or codify (Carmel & Tija 2005). It is fuzzy knowledge learned from practice, exposure, and experience – the "know–how." It is also the "know–who" of social relations. Much of the tacit knowledge can only be transferred through learning by doing, through "show–how," as when a novice engineer learns on-the-job through mentoring and coaching by a senior engineer.

A specific activity that embodies a great deal of tacit knowledge is that of Software/IT Architects. Software/IT architects are the people who oversee design at the highest level and orchestrate new product development from conceptualization to implementation to maintenance/support. Architects are not specifically trained in universities for their jobs. Rather, they learn on the job following a career path, in the case of software architects, from programmer or systems analyst to project manager to architect of a system or the integration of multiple systems. Through experience, the architect comes to understand how to achieve balance between the system's utility, cost, and risk, and how to ensure that the system meets the requirements of its intended use today while also designing a robust system capable of meeting the needs of tomorrow (ANSI/IEEE Std 1471-2000; Kruchten et al. 2006). Consequently, the literature identifies the IT architect function as one of the least likely or last to be offshored (Zwieg et al. 2006). IT architects work upstream in the new product development process and are closely tied to design innovation, which might also make their work strategic.

Because tacit knowledge is difficult to transfer it is typically a Stay factor. However, we hypothesize that once tacit knowledge is present in an offshore location, it will begin to pull similar knowledge work offshore.

Table 1 The Pull and Stay factors introduced and discussed in this paper

Factor	Is it an Offshore PULL Factor?	Is it an Onshore STAY Factor?
1. Tacit knowledge	Yes	Yes
2. Technology transfer policy	Yes	Yes
3. Control of standards	Yes	Yes
4. Critical mass of activities	Yes	Yes
5. Proximity to clients and to lead clients	Yes	Yes
6. Synergistic operations	Yes	Yes
7. Eagerness	Yes	–
8. Freshness	Yes	–
9. Crown jewels	–	Yes

Technology Transfer Policy. Some nations require foreign firms to transfer technology in exchange for market access. Such transfer might occur through licensing, joint ventures or joint R&D. Usually this is only a factor for countries with big emerging markets such as China and India, or lead user markets such as Japan and Korea, who are in a good bargaining position to set formal or informal requirements of foreign firms in exchange for market access.

One of the major factors in the award of a joint venture with China's Shanghai Automotive Industry Corp (SAIC) to General Motors was its agreement to advanced and continuous technology transfers through state-of-the-art technology, joint R&D projects and training (DFI International 1997; The Economist 1999). More subtle pressures are used in other industries. For instance, Microsoft provided access to its source code to several governments, including China, in response to security concerns (CNET 2003). It also set up an R&D center in Beijing, at least in part to improve relations with the Chinese government (although the Chinese government has failed to prevent large-scale software piracy and continues to support open source alternatives to Microsoft products).

By contrast, home countries also can restrict technology transfer. For instance, Taiwan has prohibited its companies from moving their most advanced chip manufacturing processes to China. The U.S. restricts transfer of many technologies to various countries including China for national security reasons. In these cases, government policy inhibits the offshoring of sophisticated activities.

Thus, in summary, technology transfer policy is a Pull and a Stay factor.

Control of Standards. Standards represent any kind of codified technical specification or methodology that are used in IT/software. These include, for example, the CMM, ISO and SOX standards for IT/software development processes. Standards are generally an enabler, allowing work flows to move around the globe more easily since less of the knowledge is tacit or proprietary. However, in some cases, a firm or a country exerts control over a standard, thus altering its neutral enabling characteristics.

Another type of standard is a product standard, which can be a source of competitive advantage for software companies (Shapiro & Varian 1998). For example, Microsoft controls the key software standards for PC operating systems, and has tried with varying success to extend its control to servers, PDAs, media players, smart phones and the Internet. Those who create and control standards for an industry are in a position to reap huge financial benefits as shown by Microsoft's profit margins and market capitalization, or the performance of other standard setters such as Oracle, Adobe and Cisco. Such companies tend to keep development and upgrading of standard-setting technologies in-house and onshore in order to maintain tight control.

Control of Standards is generally a Stay factor.

However, there are some areas where it is becoming a Pull factor. China has focused on several standards around which it has also built or attracted expertise. For example, the TD-SCDMA (Time Division-Synchronous Code Division Multiple Access) technology, developed by China in conjunction with Western companies, is an official standard for 3G mobile telephony in China. Firms that wish to develop 3G products and services for this market will likely have to locate R&D in China.[1] In a different domain, Indian firms have been faster than U.S. companies to adopt process standards such as ISO and CMM, a factor which has certainly helped pull more software development offshore by signaling high quality to the marketplace. Moreover, in recent years, Indian organizations have begun to exert increasing influence on these standards setting bodies.

Critical Mass of Activities. When activities are highly interrelated it is better to colocate them and to decouple them from other activities that are only loosely related. The colocated activities sometimes become a critical mass of activities that draw new tasks to that location like a magnet. We see two layers of such critical mass: the first is internal to the firm; the second is in the firm's technological cluster.

Certain activities are colocated because they are highly interrelated and require proximity for effective performance. Product design is such an activity as it requires the frequent interaction of specialists from marketing, market analysis, industrial design, engineering and product management. A critical mass of such specialists are colocated because proximity matters, especially for such complex tasks involving multiple disciplines. However, many units perform focused and isolated tasks that require relatively little interaction and can function effectively in distributed offshore locations. This is called decoupling.

For example, some activities in offshore development centers (ODCs) are easier to do independently once they reach a steady-state. Once the work is shifted there, it is unlikely to be moved again and unlikely to be shared with other locations. Thus, once there is a critical mass around some activity, it develops its own force of attraction in pulling similar activities to the same location. Of course, this notion is common to many forms of work. In IT and BPO (Business Process Outsourcing), we see this manifestation of critical mass with the acceptance of the "Shared Services" concept. Shared Services often involves the centralization of services enabling some economies of scale. Firms now accept that centralization of services is better and cheaper than the decentralized model of yester-year.

The other tier of critical mass is known as "Cluster Effects," best illustrated by Silicon Valley (Porter 1998; Saxenian 1994). Bangalore is now a software and services cluster. Beijing's Haidian district has now developed cluster characteristics in software. Clustering of firms together means that there is an agglomeration of suppliers of inputs or complementary services such as legal, financial and marketing services. The firm actually benefits by being next to other firms that are similar to it. Thus, location is determined by other firms in the region in order to tap human resources, supplier firms, and infrastructure.

[1] It is worth noting that Motorola has established large R&D and engineering activities in China, partly to develop products for the world's largest mobile market – whatever the standards may be, so such standard setting efforts by the Chinese government might not have much of an additional pull effect on Motorola.

Critical mass of task activities is a Stay factor, however once a critical mass does move offshore, it becomes a Pull factor.

Proximity to Clients and to Lead Clients. Clients demand proximity, face-to-face interaction, and relationships. Providers need to be close to the clients to meet, shake hands, have lunch and work together. Lead clients are the most innovative clients and those from which the firm learns the most (von Hippel 1994). These are the clients that help make the technology firm successful.

Proximity to Clients–and to Lead Clients – is a Stay Factor because until just a few years ago there were few clients of significance in many of the offshore markets. The lead clients – the most innovative – were onshore in developed countries (mainly the G7 nations). Offshore nations (developing/emerging) did not have sophisticated/discerning consumers; they had little middle class buying power; their corporations were relatively rudimentary in their use of technology, their ability to integrate technology, their size, and their resources.

Given this context, foreign software firms had to go abroad to reach lead clients. Indian firms grew by serving clients in the U.S. while Israeli software startups succeeded by entering the American markets with their product innovations.

However, the context is different now. The largest global firms are dispersed around the world with a very large presence in India and China. Also, there are large local companies in both India and China, as well as in some other emerging countries. These lead clients are now a *Pull factor* in the developing world as well as in the developed world.

Synergistic Operations. Many firms have some kind of operations already located in one or more places around the world. These may be sales, distribution, manufacturing or even design and R&D. As a result of such operations, the organization develops knowledge of the location's human capital, work culture, infrastructure and government policy.

The location draws other activities especially when there are synergies between activities. Thus, an onshore manufacturing site which has been very successful in recruiting local staff for its own IT shop, might become attractive as a place for performing the corporate IT function. *This is a Stay factor*.

Conversely, an engineering design center established to collaborate more closely with its manufacturing outsourcer, might develop a cost-effective talent pool of CAD software engineers. In this case, there is a *Pull* of software activities from headquarters to the offshore location. In some cases, the captive offshore operations may be totally managed by locals who also have personal connections (family, schoolmates) and deep understanding of the local culture and institutions that uncover additional opportunities. The local managers may develop an affinity to the offshore location and lobby for expanding activities there. Further, some of the local managers might have advanced to high-level positions in the headquarters location where they become advocates for moving work to their home country based on its comparative advantage. The movement of manufacturing to the Asia–Pacific region and China, as well as the movement of software to India, is credited in part to their large diasporas which functions in this manner.

Thus, existing synergistic operations can be either a Pull factor or a Stay factor.

Eagerness. Once you expose an elite set of engineers to the action and excitement and challenge of the global leading edge in software, they will not be content to settle for mundane tasks. They will demand more. They will "pull." *Eagerness is a Pull factor*. The young engineers have the confidence of working with global firms; they will demand the most challenging tasks that they now feel entitled to (Saxenian 2006). Also, local managers of MNC subsidiaries desire to increase the size and scope of their own operations, and will begin to compete within the company to host different activities.

Freshness. Companies seek fresh perspectives in design: in creativity, in innovation, in ideas. For example in the auto industry, several Japanese companies sought fresh perspectives by setting up design centers in trendy Southern California (Honda in Torrance, Isuzu in Cerritos, Mazda in Irvine, Subaru in Cyprus, Toyota in Gardena). The trendy clients are there, but so are the trendy designers. By 2006 there were 3,600 employed in Japanese automakers' design centers in the U.S. (Moavenzadeh 2006).

Another approach to freshness is to rely on acquiring innovation offshore. Israel is an incubator for innovative new software technologies in its hundreds of start-ups. These are then acquired and absorbed by foreign multinationals (De Fontenay & Carmel 2004). Cisco, Intel, and IBM have each acquired several Israeli start-ups in recent years.

Almost all software has to be localized to local language and culture. Situating localization in the target market allows firms to better customize products to the local markets, particularly the large and more promising markets, such as India and China. This is the case with Microsoft's significant presence in China, with its development centers in Shanghai and Beijing which devote significant resources to Chinese language scripts and other local needs.

Thus, the quest for freshness is a Pull factor.

Crown Jewels. Companies tend to keep "high-end" tasks at home (onshore) as well as in-house to maintain their competitiveness. These "high-end" tasks have many labels, such as core activities, proprietary activities, sensitive activities, and leading product design. Regardless of label, they usually are key to the company's competitive edge. Companies are reluctant to let the crown jewels from under their immediate watch whether these jewels are outsourced onshore or outsourced offshore or even in-housed (in a "captive" center) offshore. The lack of effective legal protection of IP in many offshore nations.is a major concern to firms with significant IP. *Thus, the crown jewels is a Stay factor*.

The crown jewels might involve tasks that are easy to copy or emulate if made public, but they might also be creative, innovative, and research-oriented, or require very broad knowledge and experience. For a software product company such as Microsoft, Oracle or SAP, its software code base is its "crown jewel." Some large user corporations also have some IT which is in this category (e.g., Amazon), but most IT applications at end user firms (e.g., banks, insurance firms) involve relatively few activities which are proprietary. Such companies are not hesitant to offshore as these applications are not the firms' crown jewels.

There are few companies that have not sourced some innovation or crown jewels offshore. But there are dangers, as pointed out by Engardio and Einhorn (2005):

> "Start with the danger of fostering new competitors. Motorola hired Taiwan's BenQ Corp. to design and manufacture millions of mobile phones. But then BenQ began selling phones last year in the prized China market under its own brand. That prompted Motorola to pull its contract. Another risk is that brand-name companies will lose the incentive to keep investing in new technology. "It is a slippery slope," says Boston Consulting Group Senior Vice-President Jim Andrew. "If the innovation starts residing in the suppliers, you could incrementalize yourself to the point where there isn't much left."

Even companies where software is a crown jewel exhibit apparently contradictory behavior. For example, there is significant offshoring of R&D among independent software vendors. It is quite difficult to estimate the magnitude of this phenomenon, but it was estimated that 5–15% of American software product R&D was offshored in 2003 and such offshoring was forecast to rise to 25–30% by 2007/2008 (Martin 2003). Some of the companies which are offshoring software product R&D are the largest U.S. technology firms. This may not be so contradictory however, as significant amounts of software development may not involve the crown jewels. Also, just moving development offshore does not necessarily put core technologies at risk if it is kept in-house and steps are taken to prevent IP leakage.

3.1 Pull/Stay Factors in the Life Cycle

One method for analyzing which activities are suitable for offshoring is to use the software development life-cycle (see Table 2). Recall that the economic *Pull Factor* – low costs – tends to first pull lower-value activities such as coding and testing offshore.

Table 2 Importance of factors (*pull/stay*) to selected life cycle stages

	Requirements and architecture	Design	Coding and testing	Integration
Typical positions involved→	IT architect; Systems analyst	Database designer	Programmer; QA engineer	Programmer; QA engineer
↓Pull/Stay factor				
1. Tacit knowledge	High	High	Low	High
2. Technology transfer policy			N/A	
3. Control of standards	High	High	High	Medium
4. Critical mass of activities	Low	Low	High	Low
5. Proximity to clients and lead clients	High	High	Low	High
6. Synergistic operations	High	Medium	High	High
7. Eagerness	High	High	Medium	Low
8. Freshness	High	High	Low	Low
9. Crown jewels	High	High	High	High

Yet, most *Pull/Stay* factors are strongest at the early stages of the value chain – *upstream*. The important exception is one *downstream* activity – the integration stage.

In general, the life-cycle phases and offshoring are driven by two factors: proximity and location of knowledge. Proximity means the proximity to activities, to operations, to clients, and to resources. Knowledge refers to tacit knowledge, IP knowledge, and trade secrets.

The integration stage is interesting in its effect on *pull/stay*. This stage usually requires all the IT/software components to come together at one location in order to build and debug. The one location is at the hub of activity – usually in the company's home country or at the site of a major client. Hence, we see the Indian-based firms send key personnel to the client site at the end of the development cycle.

3.2 Examination of the Pull/Stay Framework with Exploratory Data

Here we use our exploratory field data to illustrate the framework. (The data sources were described in the methodology section). For each of the nine *Pull/Stay* factors we present data. Some of the data are supportive, some unsupportive, and sometimes they are contradictory.

Tacit Knowledge (TK). In our software interviews in China, we heard conflicting claims about the transfer of TK. Using IT architects as an example, some sources argued that architects are still coming from the U.S., that the work is still specified abroad and that the Chinese generally don't have such experienced people. Others point out that the Chinese are developing such TK. Part of the discrepancy may be definitional. As with many titles in IT, there is no IT architect title per se, the definition is fuzzy and the function does not require a particular license or degree.

At the Chinese R&D site of a giant American tech firm our source stated that TK is still in the USA; that the "big picture" is not here [in China] and that "they're still quite young [in China]." Another source at a different firm argued that there were no software architects in China because engineering education was theoretical and software engineers came without practical skills from universities–as opposed to the more practice-minded IITs[2] in India. At another firm we heard that Chinese software engineers don't know about business or business processes because they are not trained in business and that, anyway, Chinese business is all based on *guanxi* (interpersonal connections and relationships, particularly with people close to the party and the government).

Technology Transfer Policy. Interestingly, we have no data about this factor pulling software development offshore in our exploratory dataset. However, IBM and other leading PC companies were required to engage in joint ventures with local Chinese computer firms in the early 1990s, and Microsoft was required to engage

[2] IIT – Indian Institute of Technology, the top-tier technical universities in India.

a domestic firm to create the Chinese version of Windows for sale in the PRC. We suspect that this *Pull* factor surfaces in narrow cases at particular times, and is becoming more subtle as blatant requirements are likely to be challenged in the WTO or draw retaliation from other governments.

Prohibitions on technology transfer are becoming less effective as well, as they are seen to put national companies at a disadvantage with foreign competitors whose governments do not have similar restrictions. For example, if the U.S. government prevents U.S. companies from transferring technology to China, Japanese or European companies may do so.

Control of Standards. It appears that this factor surfaces only in particular technologies. For example, China required US-Tech's embedded software to use the Chinese crypto standard. As a result, US-Tech had to contract with a local Chinese company to write the crypto code.

Firms that control product standards may be hesitant to move development offshore for IP protection reasons. On the other hand, if they wish to establish their standards in other countries, they may need to do development in those countries in order to create products that will succeed in those countries, or work with local partners who will do so. So for instance, Qualcomm worked closely with Korean cell phone makers to establish its CDMA standard in that country, and Cisco worked with Japanese hardware manufacturers to establish its proprietary operating system in Japan.

Critical Mass of Activities. Our interviews with Chinese IT Services firms that manage offshore development centers (ODCs) reveal that, as with the Indian ODCs, once tasks migrate into those centers, the task managers become independent and tasks are unlikely to be moved again and unlikely to be shared with other locations. At that point, due to cost advantages, similar tasks tend to follow offshore.

At US-Tech, which develops some of the world's most complex computer chips, our interviewee emphasized that such tasks required complicated collaborations and multiple owners. These characteristics drive colocation in the U.S. Hence, this is a *Stay factor*. More specifically, US-Tech likes to colocate the (embedded) software with the silicon designers. Why keep/move software close to hardware design? This does not hold for all hardware activities, but, in the words of our interviewee: "[the firm is….] really focused on collocating software that directly controls the hardware, such as device drivers as being coresident with the silicon team, creates better communication, results in innovations on the silicon/software boundary, and makes initial bring up and debug go faster."

US-Tech has significant software development in India. In the case of India, there seems to be some shifting in critical mass. It used to be that all new system on chip (SOC) designs were done in the U.S. but not anymore. The India design center now has hundreds of software engineers. And now there is colocation of silicon engineers and software engineers.

Also, for US-Tech, India now offers cluster effects – agglomeration effects of supplier networks: "[We have…] ability to engage with subcontractors to augment our team to handle new projects, and [we have an] abundance of third party software partners that can provide middleware, etc. on top of our products."

We heard a contradictory piece of evidence in China. One of the interesting developments, globally, is the production of mobile telephone handsets. For these handsets, companies tend to disperse the production of the three main components: the hardware, the embedded software, and the user interface. Our source in China, for example, developed embedded software for mobile phones and was emphatic that there were no advantages to colocating hardware and software.

Proximity to Clients and to Lead Clients. As expected, this is becoming evident in India and China. For China-based providers, many of the clients are not offshoring to them, as in the India model. Rather the clients are MNCs in China: these are the foreign firms (e.g., American) that require local support for their large presence in China – in Beijing, Shanghai, Hong Kong, Shenzhen, and elsewhere. Restated – all the most advanced firms in the world are operating in the country.

Two examples support this point. First, there are now many Indian firms in China. Our interviewee at one of these companies noted that the firm sees China as a pilot roll-out site for many clients. Second, Microsoft China increasingly works not with Redmond or other American firms, but with Chinese startups.

On the subject of lead clients, we hear some confirmation from US-Tech in embedded software, that lead clients are now in Japan and China, and more recently, also in India. We note that a precedent in hardware may soon apply to software: In hardware, U.S. suppliers of manufacturing and measuring equipment such as Tectronics have moved some R&D to China to be close to their clients, which are Taiwanese firms doing original design manufacturing for U.S. PC companies. So who then is the real client? The U.S. or Taiwanese firms? These supplier firms have moved to China because they need to work closely with the Taiwanese ODMs to design new equipment that might be needed 9–12 months down the line.

Synergistic Operations. In our interviews and the literature, we found examples of existing offshore operations pulling other related activities. Dell provides an interesting Pull chronology in Brazil and Taiwan, that we describe here. We have early indications that similar pulling is now happening in India.

Dell established a production facility in Brazil in 1999. This allowed Dell to compete in this market due to cost savings obtained from tax incentives, tariffs savings, and local economic benefits. Three years later, Dell opened its Global [Software] Development Center (GDC) in Brazil, which is responsible for supporting corporate software development that is used in-house. Rather quickly the GDC began to lead in some activities. For example, the Brazil GDC was the first of Dell's offshore facilities to attain the SW-CMM Level 2 standard (Bothelo 2003). In 2004, Dell announced it would be expanding its services in Brazil to offer specialized consultancy in migration, in infrastructure and in consolidation of server and storage, training and certification.

Similarly, Dell established a design center in Taiwan in 2003 to work with its ODM contractors on engineering problems related to notebook development and manufacturing (Digitimes 2003). The reason was to be closer to where the problems occurred, to be able to see the problems and resolve them faster, and to be able to use the ODM's testing facilities rather than duplicate them in Austin. In addition, being close to the ODM enabled Dell to build deep relationships between engineers.

Perhaps more important, the offshore location enabled Dell to combine lower labor costs with distinctive skills. Low wages allowed Dell to hire more middle managers who could then devote more time to building the skills of their employees and to improving their processes that would be economical in the U.S. These greater capabilities have enabled Dell's design center to pull more work from Austin to Taiwan and move up to server design. A recent news article indicates that the design center has grown from about 100 engineers in 2003 to over 500 in 2005 and is slated to increase further (Digitimes 2005).

Eagerness. Our interviews in India and China indicate that another *Pull factor* results from the upgrading of individual capabilities through experience working on multiple projects with multinational firms. As their capabilities increase, software engineers become eager to take on higher level activities and they proactively seek them, either inside the firm or outside in their own enterprises.

In our interviews in India with engineers in ISVs we encountered this type of behavior, though still in its early stages. One lead Indian engineer working at a major American firm in Bangalore declared again and again that he and his colleagues wanted more of the firm's leading tasks, which were currently being conducted onshore in the U.S. They were confident in their abilities, felt entitled to take on more complex projects and lobbied higher level management to transfer the work.

Another Indian software engineering manager from another well-known American firm in Bangalore quit his job because he was not given enough challenges. Given his contacts and experience, and the liberal availability of venture capital, he was able to start his own firm in India. Thus, some of this *Pull* happens inside the existing chain, as with the first engineer, while some happens outside the chain as with the second engineer.

Finally, we interviewed managers in one MNC subsidiary in Australia that was competing aggressively with the company's Singapore subsidiary to run a new data center, and with a Hong Kong subsidiary for a customer support center. In this case, the eagerness of managers in each country to expand the scale and scope of activities, and the confidence they could do so, was a *Pull factor*.

Freshness. We have no data about this factor pulling offshore in our exploratory dataset. We suspected that this factor surfaces in narrow cases.

Crown Jewels. Our field interviews indicate that a firm's crown jewels might be in hardware, software or both, as in the case of embedded software on a chip. At US-Tech, the hardware division is concerned with protecting the firm's technology secrets. Hence the leading fabs and design centers are kept in the U.S. At StorageCo the core technology is not only the tiny recording head for the hard drive, but also the software for firmware. R&D, design, development of the hardware and software, are all kept in-house onshore, but manufacturing is done in-house offshore at low cost locations.

At other American-based hardware technology firms (i.e., Printco, PDAco, MP3co), the crown jewels are increasingly in software. The software controls not only the operation of the hardware but also its interface with other technologies where these controls operate as part of a complete system. Software is also a core technology because it allows the hardware products to be customized for different uses and markets without the need to build separate products for each use or market. Thus, the entire software development process is kept in the U.S. to protect IP and corporate knowledge while hardware production is outsourced offshore to low cost locations.

Thus, there is clear support in all of the interviews for the hypothesis that Crown Jewels are a *Stay factor*.

3.3 Application of the Framework to Life Cycle Stages

Although the integration phase is downstream in the software development life cycle, it is affected by most of our *Stay factors*. Integration requires some of the highest skills, proximity of lead designers, and proximity to the client. Integration seems to continue to have strong *Stay factors*.

At US-Tech the integration stage usually requires all the software components to come together at one location in order to build and debug (including "bring-up"). While some integration, for some products, is conducted in all four offshore locations, India is most interesting. The senior executive at US-Tech said: "Software bring-up is done where the silicon team exists. For certain product lines, that is in India. It's [now] more a function of having a senior team in place in India [whereas before we did not]."

Another example of integration is in the following Japanese (onshore)/Chinese (offshore) relationship in which integration comes together onshore. C-Soft is one the major Chinese software firms. A number of firms, including Japanese firms, outsource mobile telephony software to C-Soft. C-Soft found that it could not test the telephony software that it developed in China. Rather, it had to send its engineers to Japan for final test for networks not available in China.

Our field observations about the integration phase and offshoring are a modification to Khan et al's (2003) notion that downstream activities are most likely to be offshored. It appears that rather than activities representing a clear progression with downstream activities offshored and upstream activities kept onshore, the distribution is U-shaped with some activities at both ends of the life cycle needing to be done onshore. This is closer to Stan Shih's "smiling curve" which shows that higher value activities are upstream and downstream, with lower value activities in the middle (Dedrick & Kraemer 1998).

4 Synthesis and Summary

Our exploratory analysis through case studies and interviews supports the premise that noneconomic factors are important influences in decisions about the offshoring of software development activities. The analysis also provides preliminary support for the hypothesized relationships. Specifically, the findings provide support for

Crown Jewels as a clear STAY factor whereas two other factors are clearly only PULL factors: Freshness; and Eagerness. The other factors show a mixed picture. While they tend to Pull activities offshore, once the activity is offshore they may work to keep the activity offshore (a new "Stay Offshore" factor).

When one looks at the offshoring of software development activities over time, there is a clear evolutionary pattern towards more higher-value activities being offshored (Fig. 1).

Whereas in 1995 only some maintenance was offshored, by 2005 more coding and maintenance activities were offshored, with some design and testing also being offshored.

We cautiously speculate that, given some of the trends we describe in this paper, by 2010, requirements, architecture and system integration activities will remain onshore at the clients' or software vendors' location. Although not shown in the figure, it is also the case that once moved offshore, these activities seldom return to the home country; they stay or migrate to another location.

This pattern in software development is worth comparing to new product development (NPD) in PC hardware (Dedrick & Kraemer 2006), which is shown in Fig. 2. The pattern of gradual movement of more and more activities offshore over time is the same, but offshoring is generally more advanced in PC hardware than in software. All NPD activities but design were offshored by 2005 and it is likely that architecture design and product planning will be done jointly with the outsourcing firms by 2010, if not completely outsourced. Then only product concept design would be performed onshore as it must be located close to the lead market (the U.S.).

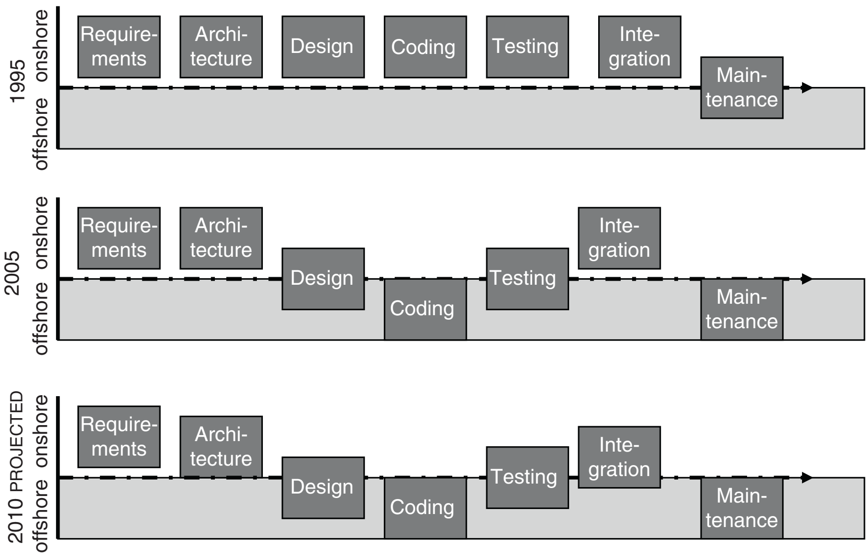

Fig. 1 Information systems and services. Adapted from Carmel and Tija 2005

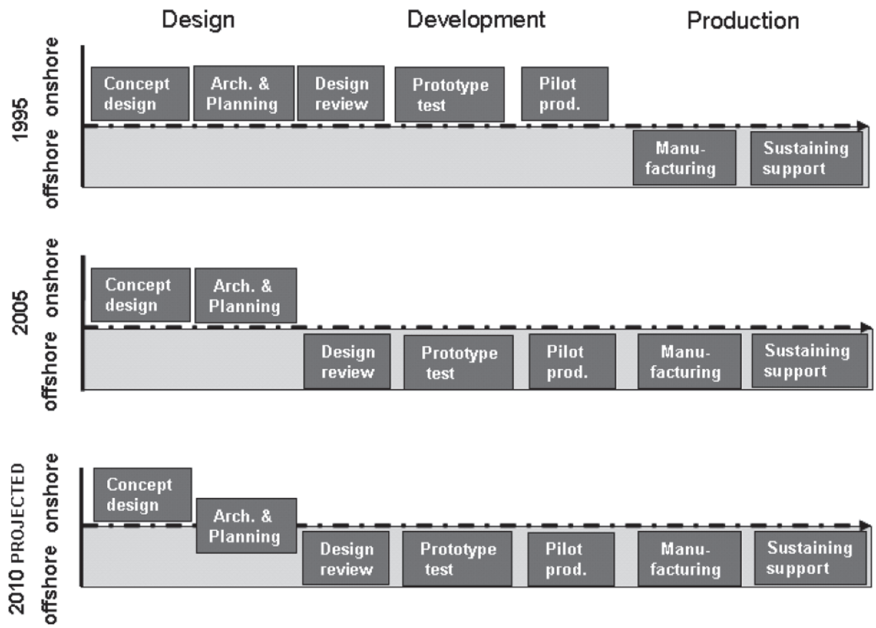

Fig. 2 New product development in PC hardware. Adapted from Dedrick and Kraemer 2006

In sum, the influence of various factors on offshoring decisions is complex, with some going both ways – acting as a staying force in one instance and pulling activities offshore in another. The dynamics or evolution of offshoring is clearer, with things seeming to move mainly in one direction from lower-level to ever higher-level activities.

Acknowledgment This material is based upon work supported by the U.S. National Science Foundation under grant number SES-0527180. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

ANSI/IEEE Std 1471-2000 (2000). Recommended practice for architectural description of software-intensive systems. accessed from http://standards.ieee.org/reading/ieee/std_public/description/se/1471-2000_desc.html

Bothelo, A. J. (2003). *The Brazilian software industry*. Working paper Instituto Gênesis, PUC – Rio de Janeiro e, Brasil. Giancarlo Stefanuto.

Carmel, E., & Tija, P. (2005). Offshoring information technology: Sourcing and outsourcing to a global workforce. Cambridge: Cambridge University Press.

Cnet (2003). China looks into Windows code. CNET News.com, September 29, 2003. http://news.com.com/China+looks+into+Windows+code/2100-1016_3-5083458.html. Last accessed March 6, 2007.

Dedrick, J., & Kraemer, K. L. (1998). *Asia's computer challenge*. New York: Oxford University Press.

Dedrick, J., & Kraemer, K. L. (2006). Is production pulling knowledge work to China? A study of the notebook PC industry. *IEEE Computer, 39*(7), 36–42.

DeFontenay, C., & Carmel, E. (2004). Israel's Silicon Wadi: The forces behind cluster formation. In T. Bresnahan, A. Gambardella, & A. Saxenian (Eds.), *Building High Tech Clusters*. Cambridge: Cambridge University Press.

DFI International (1997). *U.S. commercial technology transfers to the people's republic of China*, Part 2 (pp. 45–75). Washington DC: DFI International.

Digitimes (2003). HP, Dell expanding R&D offices in Taiwan. *Digitimes.com*, May 1.

Digitimes (2005) Dell: Procurement budget in Taiwan to surpass US$10 billion this year. *Digitimes.com*, April 20.

Economist (1999). General Motors finally won the license to establish a joint R&D centre in Shanghai, May 1.

Engardio, P., & Einhorn, B. (2005). Outsourcing innovation. *Business Week*, March 21, 2005.

Friedman, T. L. (2005). *The world is flat: A brief history of the twenty-first century*. New York, NY: Farrar, Straus and Giroux.

Greenstein, S. (2005). Outsourcing and climbing a value chain. *IEEE Micro, 25*(5), 83–84.

Khan, N., Currie, W. L., Weerakkody, V., & Desai, B. (2003). *Evaluating offshore IT outsourcing in India: Supplier and customer scenarios*. Paper presented at 36th Hawaii International Conference on Systems Sciences (HICSS), January 6–9, Big Island, USA.

Martin, K. (2003). *Research and development goes offshore*. META Group News Analysis, September 12. Corporate report.

Moavenzadeh, J. (2006). *Offshore automotive engineering: Globalization and footprint strategy in the motor vehicle industry*. Paper presented at the National Academy of Engineering workshop on the offshoring of engineering, Washington D.C.

Kruchten, P., Obbink, H., & Stafford, J. (2006). (Guest Eds.), The past, present, and future of software architecture. *IEEE Software, 23*(2), 22–30.

Porter, M. E. (1998). Clusters and the new economics of competition. *Harvard Business Review, 76*(6), 77–90.

Saxenian, A. (1994) *Regional advantage: Culture and competition in Silicon valley and Route* 128. Cambridge, MA: Harvard University Press.

Saxenian, A. (2006). *The new argonauts: Regional advantage in a global economy*. Cambridge, MA: Harvard University Press.

Shapiro, C., & Varian, H. R. (1998). *Information rules: A strategic guide to the network economy*. Boston: Harvard Business School Press.

Von Hippel, E. (1994). *The sources of innovation*. Oxford: Oxford University Press.

Zwieg, P., Kaiser, K. M., Beath, C., Bullen, C., Gallagher, K., Goles, T. et al. (2006). The information technology workforce trends and implications 2005–2008. *MIS Quarterly, 5*(2), 47–54.

Appendix 1: Enabling Factors

Enabling factors are those that enable the frictionless mobility of projects and tasks to different locations around the world. Several key factors are listed here.

- *Technology*. The range of communication and collaboration technologies from e-mail to project-ware, and many more. These are now cheap and ubiquitous.
- *Standards*. Standards allow codified knowledge to flow easily to offshore locations. The standard creates such a level playing field that there is no advantage to the traditional bastions of knowledge in the USA. One can see the enabling impact of standards in the field of hardware. Taiwan was able to move from CM to

CDM in the 1980s because basic components became standardized (Engardio and Einhorn 2005).

- Global Collaboration *Methodologies*. These ease task hand-offs with distant collaboration. For example the Indian IT firms have their GDM, the Global Delivery Model, which codifies such work.
- Professional *providers and intermediaries* have emerged to oil, smooth and otherwise help in moving tasks offshore, e.g., the offshore provider Ness runs centers called Build Operate Transfer (BOT) in India for its clients. Such a structure, now increasingly common, eases the complexity of offshoring and reduce risks.
- *Ease of doing business*. Increased number of joint ventures and alliances into the new offshore locations.
- *Relationships*. Relationships formed between people and firms help smooth the transitions. Legal arrangements are easier. Now, more than a decade in the new globalization era, a network of relationships of global managers and ex-pats have been formed.

Appendix 2: Moving up the Value Chain

Moving up the value chain is a topic that has received a great deal of attention. We know that this topic is an important one because there are many terms to describe it: moving up the value chain, moving up the food chain, moving upstream. We present two such value chains in Table 3.

Table 3 Examples of moving up the value chain

Greenstein (2005) For *computer hardware* value chain	1. CM – Contract Manufacturing 2. CDM – Contract and Design Manufacturing 3. ODM – Original Design and Manufacturing
Khan et al (2003) For *IT services firms in India*	1. Body shopping 2. Offshore development and project execution 3. Establishing standards 4. Consultancy and designing the IT architecture 5. Design and product development